Voronoi 図を用いた長方形ロボットの経路探索*

鈴木 一平† 今井 桂子‡
中央大学大学院理工学研究科情報工学専攻†
中央大学理工学部情報工学科‡

概要

近年ハードウェア支援技法により作成した Voronoi 図を用いて経路の探索をおこなう手法が提案された．この手法は Voronoi 図から初期経路を求め，それに沿って実際にロボットが動作したときに衝突する部分で経路を修正するものである．本研究では，経路の修正部分に注目し，ロボットがより滑らかに動作できるような手法を提案する．さらに，提案手法を実装し実験をおこなったのでその結果を報告する．

1 はじめに

ロボットの自律移動において，障害物に衝突せずに目的地に到達することは重要な課題である．この問題は計算幾何学やロボット工学の分野で数多くの研究がなされてきた．その手法の一つにロードマップ法がある．これは，自由空間の連結性を表すグラフを作成し，その上で経路を探索するというものである．近年，このグラフを作成するためにハードウェアによって描かれた Voronoi 図を用いるという手法が提案された [1, 2]．この手法は Voronoi 図から得られる Voronoi グラフ上で仮の経路を決定し，障害物と衝突する部分でランダムサンプリングを行い，経路を修正していくというものである．本研究では，経路の修正部分に対してロボットがより滑らかに動作できるような手法を提案し，計算機による実験結果を報告する．

2 Voronoi 図

点や線，多角形などの一般図形 $A_1, A_2, \ldots, A_k$ が与えられたとき，平面上の任意の点 p に対して，図形 A_i までのユークリッド距離を $\mathrm{dist}(p, A_i)$ とする．このとき，各図形に対する Voronoi 領域は以下の式で与えられる．

$$V(A_i) = \bigcup_{i \neq j} \{p | \mathrm{dist}(p, A_i) \leq \mathrm{dist}(p, A_j)\}$$

平面の $V(A_1), V(A_2), \ldots, V(A_k)$ による分割を一般図形 Voronoi 図という．

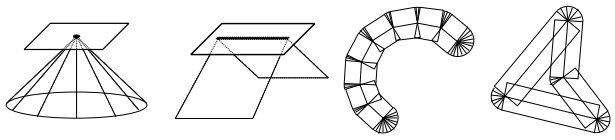
これに対して，平面を細分化し，各離散点ごとにどの領域に属するかを決定して得られる Voronoi 図を離散 Voronoi 図という．これは離散点から各サイトまでの距離を比較することによって作成することが出来る．

2.1 距離関数メッシュ

各サイトの形状によって距離関数を領域内のすべての点までの距離として定義する (表 1，図 1)．

表 1: サイトに対する距離関数の形状

サイトの形状	距離関数の形状	図
点	円錐	1(a)
直線	テント	1(b)
曲線	円錐とテント	1(c)
多角形	円錐とテント	1(d)



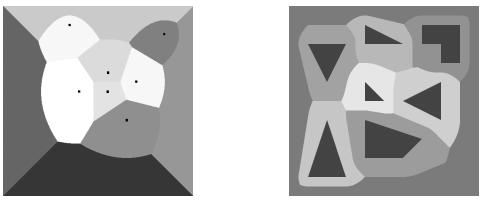
(a) (b) (c) (d)

図 1: 距離関数メッシュ

以上のように点と直線の距離関数を定義すれば，曲線や多角形の距離関数は円錐とテントの組み合わせによって定義できることがわかる．図 1 のような図形を距離関数メッシュという [2]．

2.2 ハードウェア支援による Voronoi 図の描画

定義した距離関数メッシュを Z バッファ法により描画する．これにより物体の前後関係を正しく描画することができる．距離関数メッシュを描く際には各領域を区別するために，それぞれに異なる色を持たせておく．するとすべてのピクセル上でそこに描かれている色を見ることによりそのピクセルがどの Voronoi 領域に属しているか，また，サイトからの距離がすぐに分かる．実際の描画結果を図 2 に示す．このときウィンドウの枠も直線として距離関数メッシュを描いている．



(a) 点 (b) 多角形

図 2: Voronoi 図

2.3 Voronoi グラフ

Voronoi グラフの頂点や辺は Voronoi 点や Voronoi 辺に対応する．まずはじめに，描画領域のすべてのピクセルに描かれている色を読み込み，隣接するピクセル同士を比較することで Voronoi 点，Voronoi 辺を抽出する．このようにして生成された Voronoi グラフ上で経路の探索をおこなう．

*Path planning for a rectangular robot using Voronoi diagram.
†Ippei SUZUKI, Information and System Engineering Course, Graduate School of Science and Engineering, Chuo University.
‡Keiko IMAI, Department of Information and System Engineering, Faculty of Science and Engineering, Chuo University.

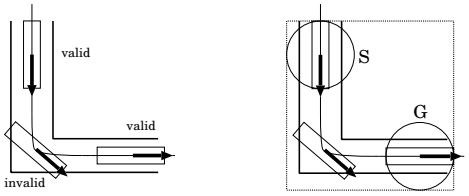
3 Voronoi 図を用いた経路探索

まず，[1] で提案されている経路探索の手法を紹介する．アルゴリズムは以下の通りである．

1. Voronoi グラフを作成する．
2. Voronoi グラフ上で点ロボットに対する経路を見つける．
3. 見つかった経路に対して，各点の接線ベクトルをロボットの向きとする．
4. ロボットを実際に動かし，衝突している部分を検出する．
5. 衝突している部分でランダムサンプリングをおこない，経路を修正する．

経路の修正

衝突している位置を invalid, そうでない部分を valid と呼び (図 3(a))，invalid な位置で経路の修正をおこなう．



(a) (b)

図 3: 衝突部分とサンプリングをおこなう空間

修正をおこなうために，invalid な部分の直前と直後をローカルな初期位置 S，目的位置 G とし，その点におけるロボットの配置によってサンプリングする空間を図 3(b) のように定義する．定義した空間内で S, G の両方からサンプリングをおこなっていき，経路を局所的に探索する．二つの経路がつながったなら，invalid な位置は valid な位置によって置き換えられ，経路が修正される．ある程度の反復で修正できない場合には，空間を広げて再びサンプリングをおこなう [3].

4 提案手法

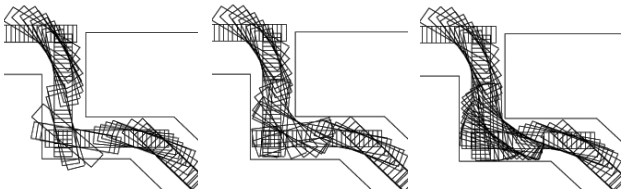
ランダムサンプリングにより修正された経路は，ロボットの位置や向きの変化にばらつきがあり，実際のロボットの動作としてふさわしくない．そこで，図 3(b) の空間内での衝突部分の修正に対して，以下のアルゴリズムを提案する．

1. 定められた空間内で障害物とロボットのミンコフスキー差をとり，コンフィギュレーション空間を求める．
2. コンフィギュレーション空間内を局所的に探索していき，ローカルな目的位置 S に近づけていく．
3. それ以上近づけなくなったところでロボットの向きをローカルな目的位置 G での向きに近づけるように微小変化させ，1 に戻る．
4. G に到達したら終了．もしロボットの向きが G での向きを過ぎたなら通りぬけ不可能とする．

この手法により，ロボットは invalid な位置において向きを単調に変化させながら進んでいく．これによって，より滑らかな動作を得ることができる．

5 計算機実験

上記の手法を実装して計算機実験をおこなった．実装には C++, OpenGL さらに GLUT ライブラリを用いた．実験結果を図 4 に示す．



(a) 修正前の経路 (b) ランダムサンプリング (c) 提案手法による修正

図 4: 衝突部分での経路修正

図 4 から，ランダムサンプリングにより経路を修正した場合にはロボットの向きの変化にばらつきが生じることがわかる．これに対して，提案手法では向きを単調に変化させ，コンフィギュレーション空間の各スライスにおいてローカルな目的位置に向かう力をかけることで，滑らかに動作させることができる．ロボットの向きの変化を図 5 に示す．

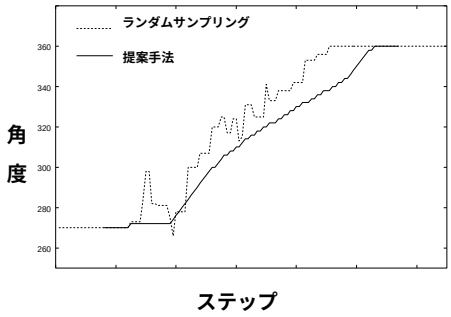


図 5: 動作中の向きの変化

6 おわりに

本研究では，[1] の手法をもとに長方形ロボットがより滑らかに動作できるような手法を提案した．また，実装して計算機実験をおこなうことにより実際の動作を確認した．さらに，ミンコフスキー差をとってコンフィギュレーション空間を求めることにより，通り抜けできるかどうかの判定もより正確にできるようにした．

参考文献

[1] M. Foskey, M. Garber, M. Lin, and D. Manocha. “A Voronoi-Based Hybrid Motion Planner,” *Proc. IEEE/RSJ International Conf. on Intelligent Robots and Systems,* 2001.

[2] K. Hoff, T. Culver, J. Keyser, M. Lin, and D. Manocha. “Fast Computation of Generalized Voronoi Diagrams Using Graphics Hardware,” *Proc. ACM SIGGRAPH,* 1999.

[3] D. Hsu, J.-C. Latombe, and R. Motowani. “Path planning in expansive configuration space,” *International Journal of Computational Geometry and Applications,* 9(4 - 5):495-512, 1999.